KDD 2018

Detecting Spacecraft Anomalies Using LSTMs and Nonparametric Dynamic Thresholding

Kyle Hundman

Data Scientist, Group Lead

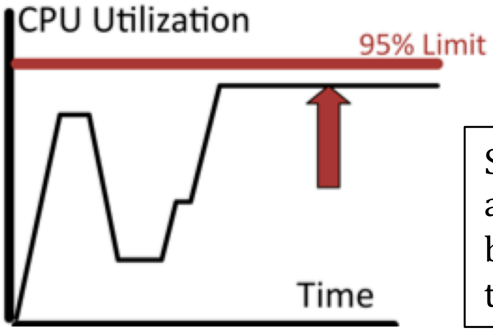
Jet Propulsion Laboratory
California Institute of Technology

Overview

- Use LSTMs to incrementally predict incoming telemetry values using recent telemetry, commands, and EVRs (event records) as inputs into a model
- Where predictions substantially different from actual telemetry values, these are identified as potentially anomalous events
    - New nonparametric method for defining "substantially different"

Motivation

- Increasing data rates
  - SWOT, NISAR = 3-5 TB daily
- Smaller missions
  - Less people (cubesats, instruments) for ops
- High volumes of testbed data
- Condensed mission operations
  - Europa Lander = 20-30 days



Simple example of anomaly that would be undetected by a threshold

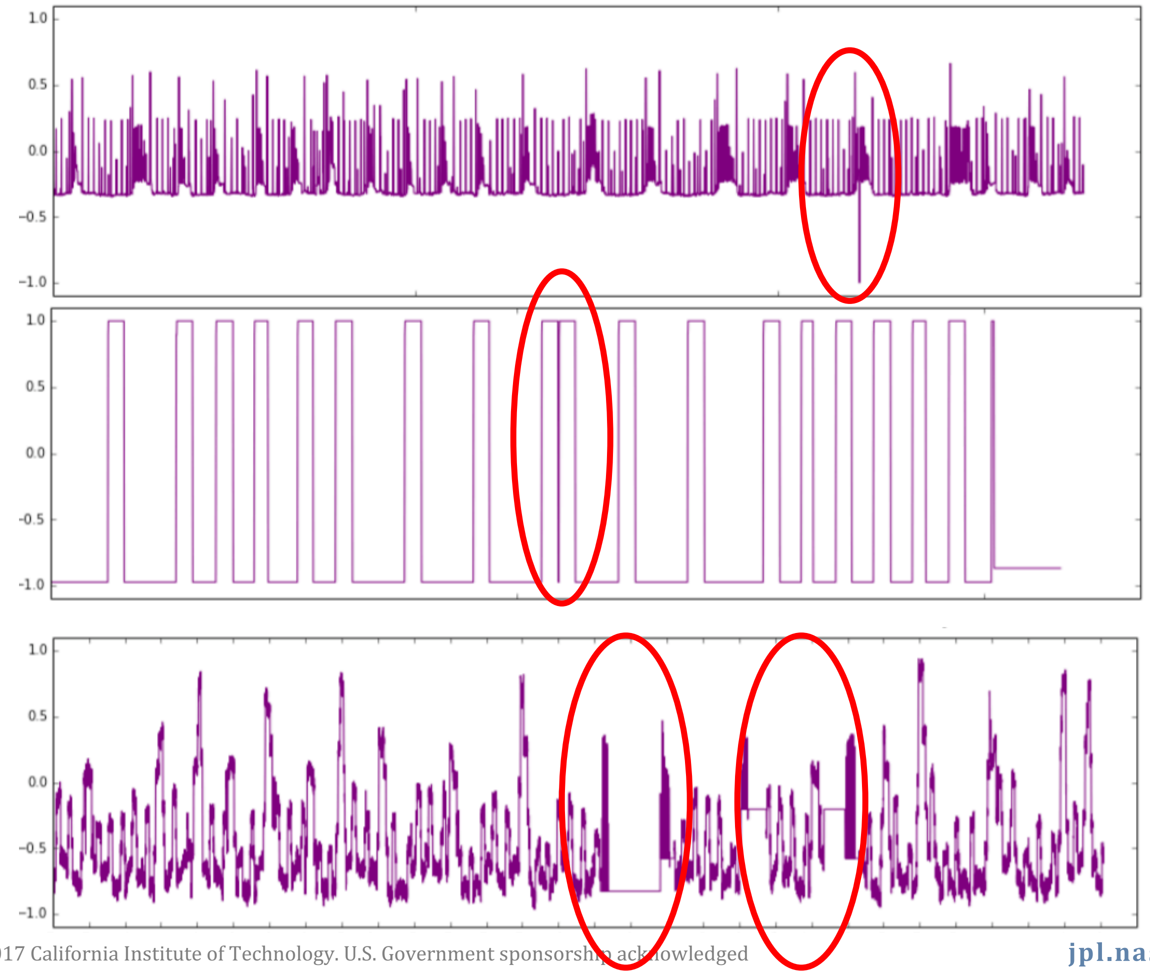
- Investigative aspect
  - Focused, prioritized telemetry review
  - Help with causal fault analysis
    - What anomalies were detected leading up to a failure?
- Thresholding, expert systems
  - Reliance on expert knowledge
  - Custom
  - Not complete
  - Accuracy
  - Appropriate limits change

~40% of anomalies in experiments are of this nature

Anomaly Categories

Chandola et al. 2007

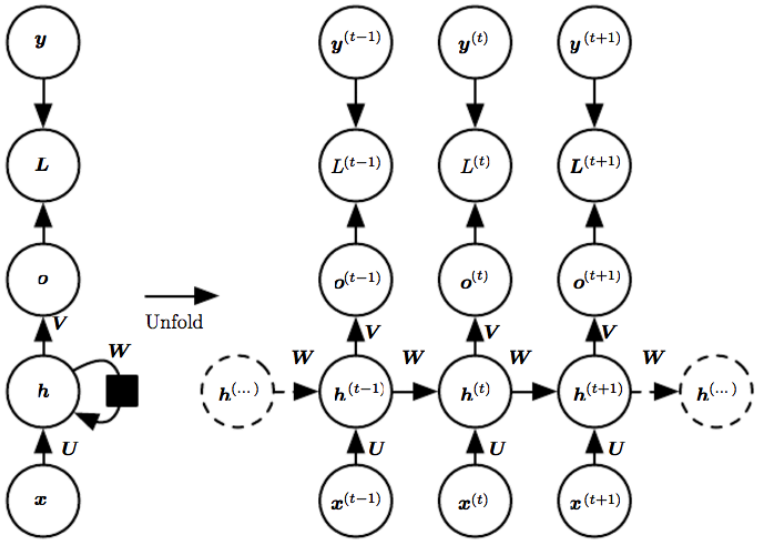
1. Point
2. Contextual
3. Collective (sequential)



Recurrent Neural Nets

- Memory (lossy summary)
- Parameter sharing
  - Extend model to apply to different lengths and generalize across time steps
    - Don't have to have separate parameters for each time value
- Recurrence
  - Always has same input size regardless of sequence length

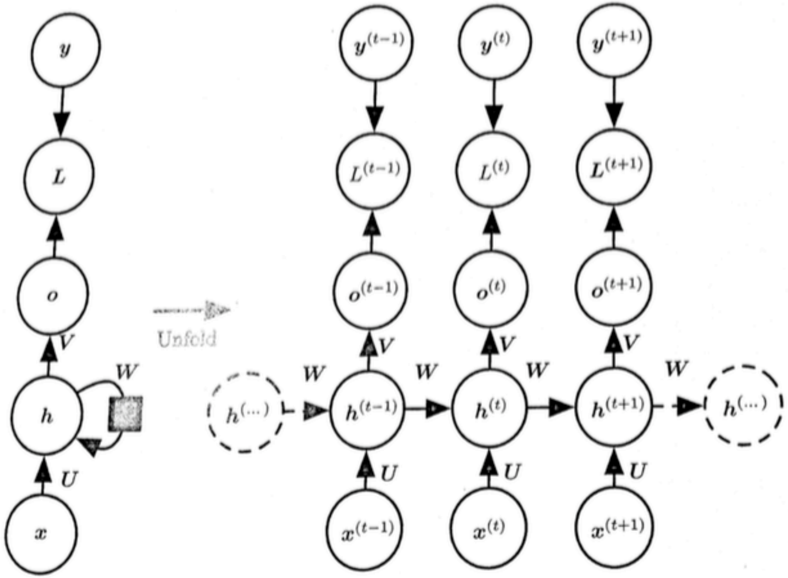
$$\begin{aligned}\boldsymbol{h}^{(t)} &= g^{(t)}(\boldsymbol{x}^{(t)}, \boldsymbol{x}^{(t-1)}, \boldsymbol{x}^{(t-2)}, \ldots, \boldsymbol{x}^{(2)}, \boldsymbol{x}^{(1)}) \\ &= f(\boldsymbol{h}^{(t-1)}, \boldsymbol{x}^{(t)}; \boldsymbol{\theta}).\end{aligned}$$



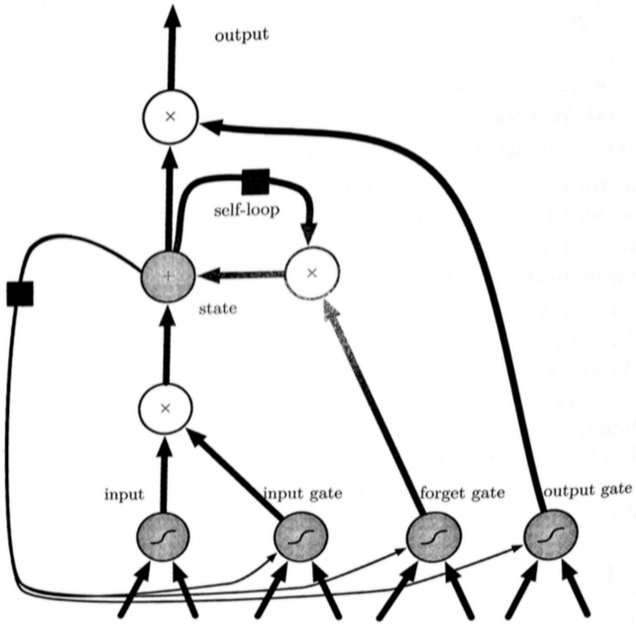
Ian Goodfellow, Yoshua Bengio, Aaron Courville, 2016. *Deep Learning.* MIT Press. http://deeplearningbook.org.

From RNNs to LSTMs (Goodfellow et. al, 2016)

RNN



LSTM



Core contribution (1997): Self-loops

Crucial addition (2000): Condition loop on context (with another hidden unit)

Ian Goodfellow, Yoshua Bengio, Aaron Courville, 2016. *Deep Learning.* MIT Press. http://deeplearningbook.org.

Formulation

$$X = \left\{ \overbrace{\begin{bmatrix} x_1^{(t-l_s)} \\ x_2^{(t-l_s)} \\ \vdots \\ \boxed{x_m^{(t-l_s)}} \end{bmatrix}, \ldots, \begin{bmatrix} x_1^{(t-1)} \\ x_2^{(t-1)} \\ \vdots \\ \boxed{x_m^{(t-1)}} \end{bmatrix}, \begin{bmatrix} x_1^{(t)} \\ x_2^{(t)} \\ \vdots \\ \boxed{x_m^{(t)}} \end{bmatrix}}^{\text{Model inputs at step } t}, \underbrace{\begin{bmatrix} x_1^{(t+1)} \\ x_2^{(t+1)} \\ \vdots \\ \boxed{x_m^{(t+1)} = y^{(t)}} \end{bmatrix}}_{l_p = 1} \right\}$$

Telemetry Values

h = historical window of errors
l_s = sequence length

$$e^{(t)} = \left[\hat{y}^{(t)} - y^{(t)}\right]$$

$$\mathbf{e} = \left[e^{(t-h)}, \ldots, e^{(t-l_s)}, \ldots, e^{(t)}\right]$$

Single-Channel Prediction

$$\mathbf{t} = \left\{[106], [107], [108], [109], [110], [111]\right\}$$

$$\begin{array}{r l}
\text{Cmd sent to Module A (T/F)} & \\
\text{Cmd received by Module A (T/F)} & \\
\text{Cmd sent to Module B (T/F)} & \\
 & X = \left\{ \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 1.40 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 1.40 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 1.40 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 1.45 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \\ \vdots \\ 1.45 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 1.40 \end{bmatrix} \right\} \\
\text{Telemetry Value} &
\end{array}$$

Same command info for every channel

$$\hat{\mathbf{y}} = \left\{[1.39], [1.39], [1.36], [1.48], [1.46], [1.41]\right\}$$

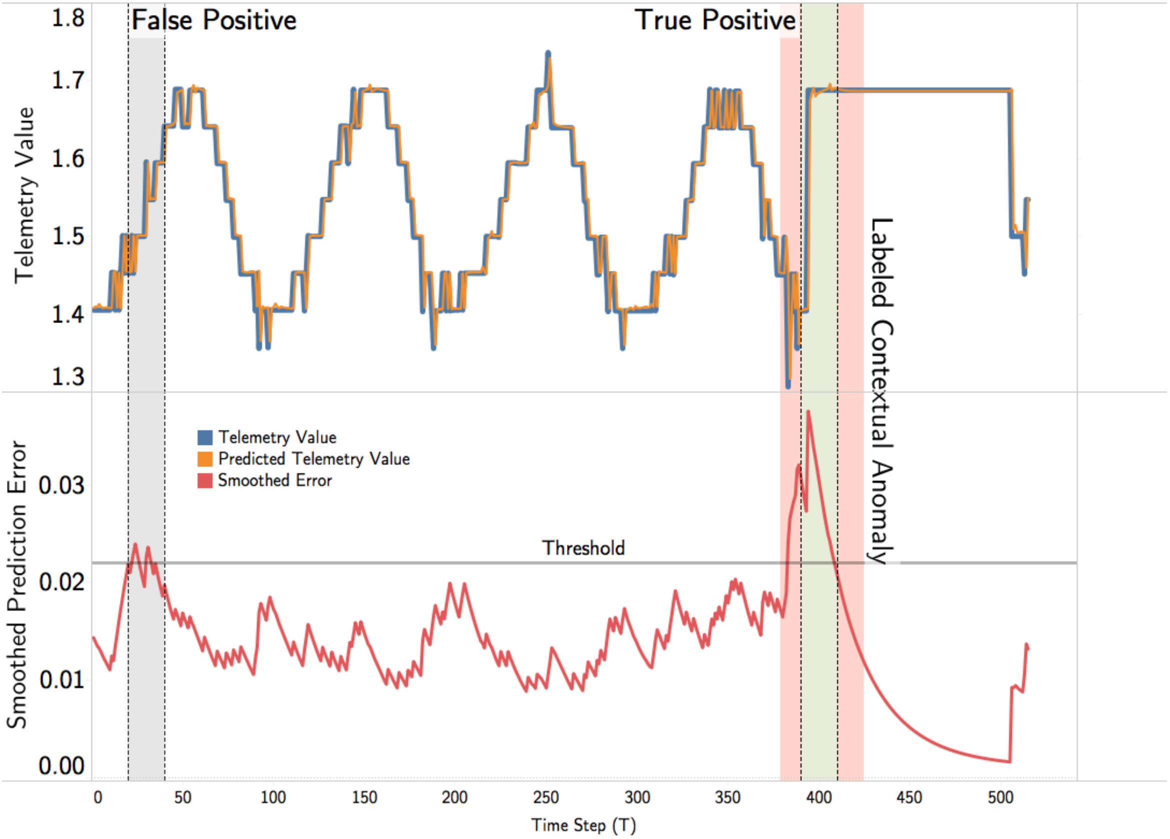
$$\mathbf{e} = \left\{[0.01], [0.01], [0.04], [0.03], [0.01], [0.01]\right\}$$

$$\mathbf{e}_s = \left\{[.016], [.014], [.015], [.017], [.015], [.012]\right\}$$

Reconstruction Errors and Smoothing

Actuals and Prediction

Raw Reconstruction Error



Dynamic Anomaly Threshold

Smoothed errors

$$\mathbf{e}_s = [e_s^{(t-h)}, \ldots, e_s^{(t-l_s)}, \ldots, e_s^{(t-1)}, e_s^{(t)}]$$

Candidate thresholds

$$\epsilon = \mu(\mathbf{e}_s) + \mathbf{z}\sigma(\mathbf{e}_s)$$

Threshold

$$\epsilon = argmax(\epsilon) = \frac{\Delta\mu(\mathbf{e}_s)/\mu(\mathbf{e}_s)) + (\Delta\sigma(\mathbf{e}_s)/\sigma(\mathbf{e}_s)}{n(\mathbf{e}_a) + n(\mathbf{E}_{seq})^2}$$

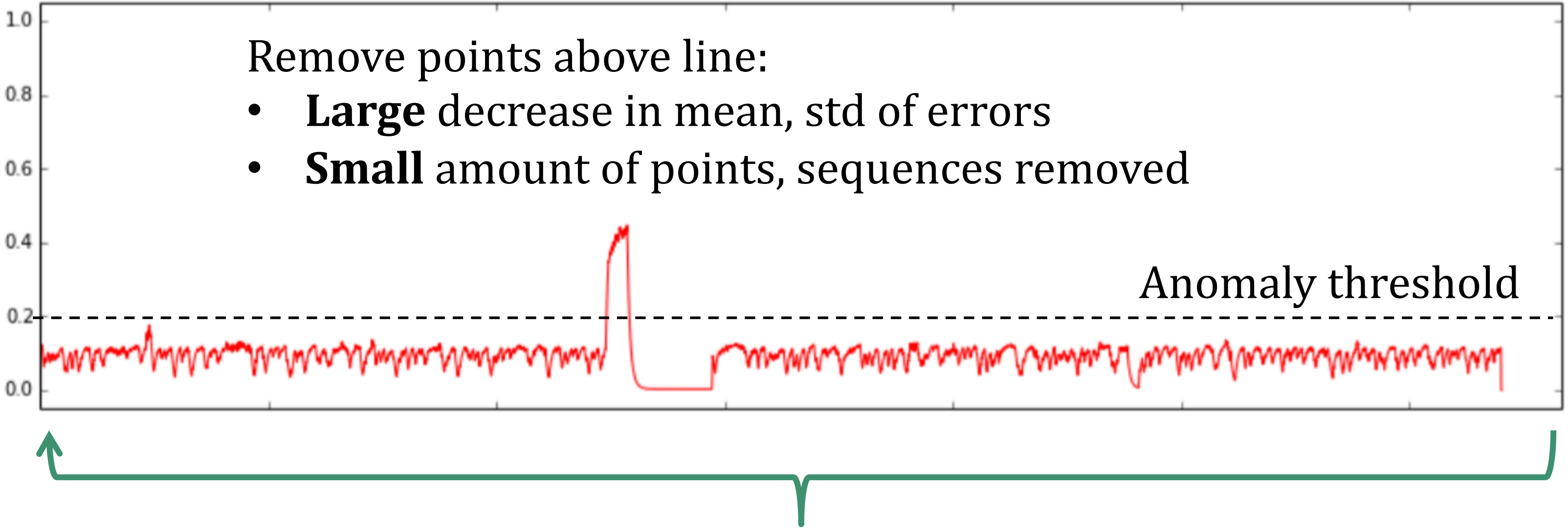
Definitions

$$\Delta\mu(\mathbf{e}_s) = \mu(\mathbf{e}_s) - \mu(\{e_s \in \mathbf{e}_s | e_s < \epsilon\})$$
$$\Delta\sigma(\mathbf{e}_s) = \sigma(\mathbf{e}_s) - \sigma(\{e_s \in \mathbf{e}_s | e_s < \epsilon\})$$
$$\mathbf{e}_a = \{e_s \in \mathbf{e}_s | e_s > \epsilon\}$$
$$\mathbf{E}_{seq} = \text{continuous sequences of } e_a \in \mathbf{e}_a$$

Dynamic Anomaly Threshold

Anomalous

Reconstruction Error

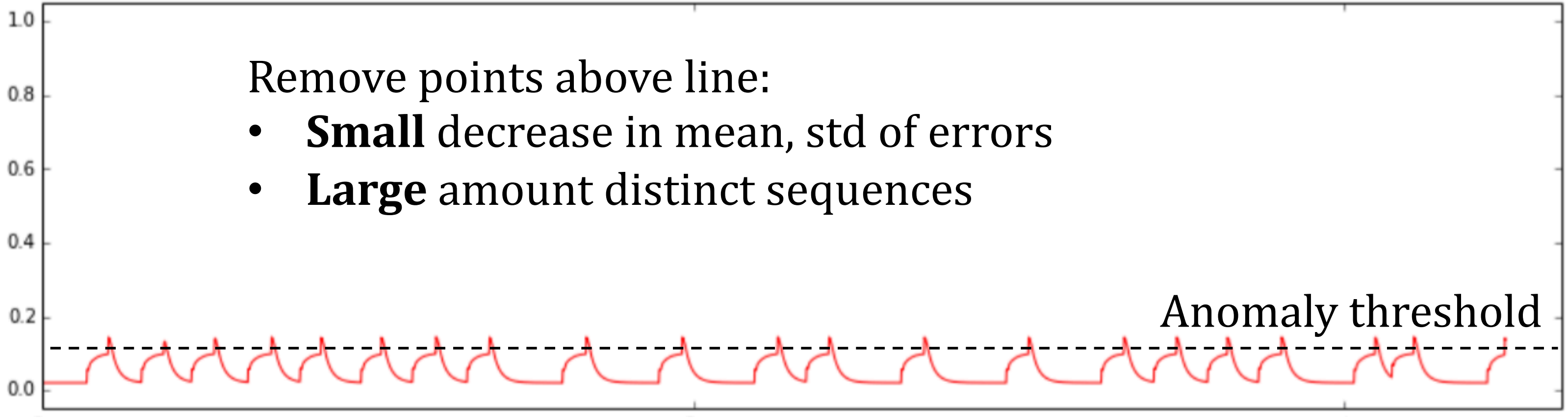


Window size (h)

Dynamic Anomaly Threshold

Nominal

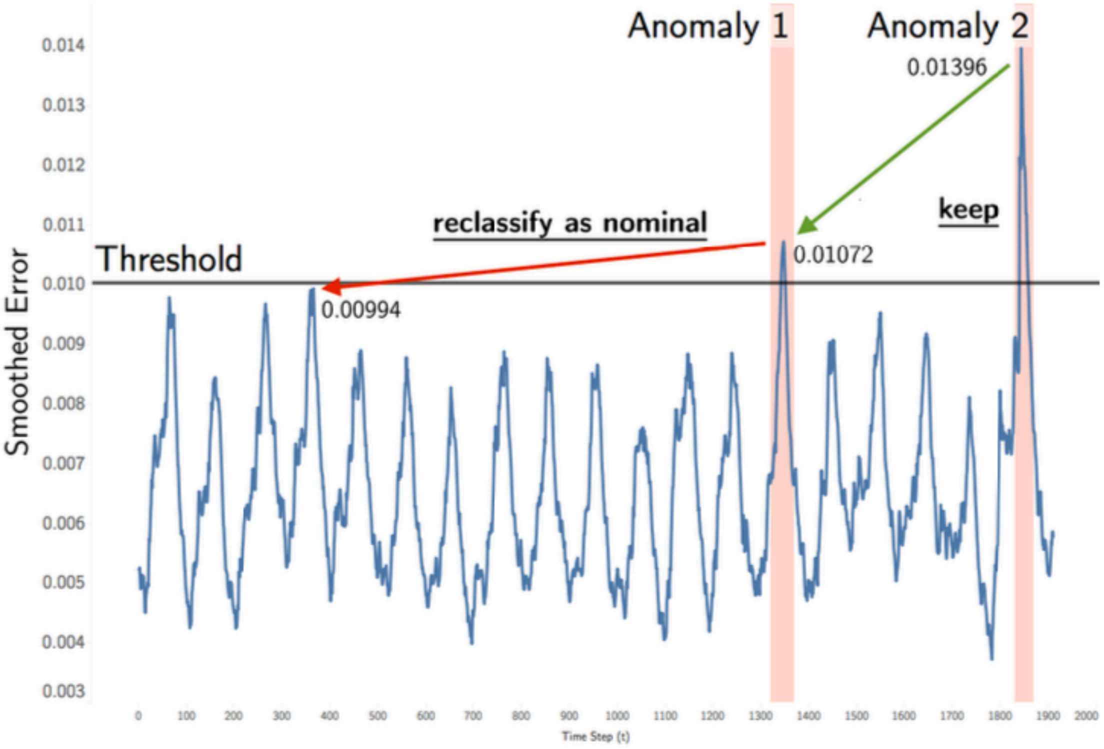
Reconstruction Error



Remove points above line:

- **Small** decrease in mean, std of errors
- **Large** amount distinct sequences

Pruning

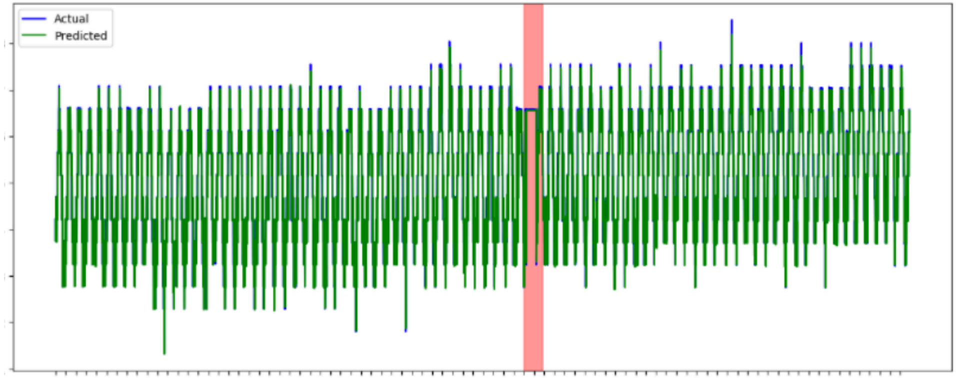


$$\mathbf{e}_{max} = [0.01396, 0.01072, 0.00994]$$

$$p = 0.1$$

Experiments – Incident Surprise, Anomaly Reports (ISAs)

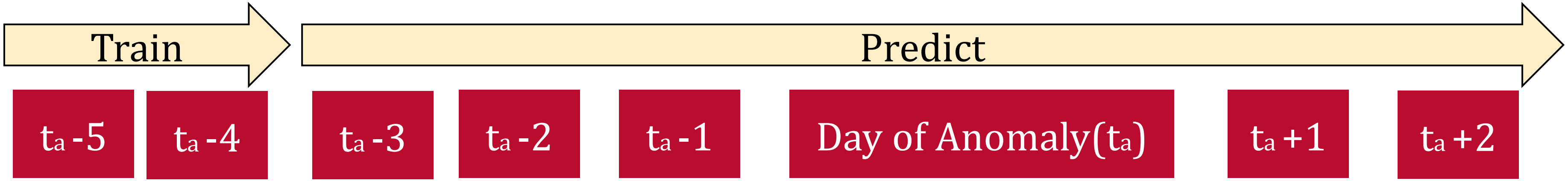
- Scraped ISAs to find mentions of telemetry channels
  - Ex. “On DOY 192, in the time range from 09:21z through 10:47z, the following channels were found to have odd constant values: A-3, A-4, A-5, A-6, G-3”



  - Labeled anomalous ranges for 112 unique ISA anomalies (MSL, SMAP)
  - Significant portion of contextual anomalies (39%)

Validation: Predicting ISAs

- Identified all Incident, Surprise, Anomaly (ISA) reports that were apparent in telemetry (EHA) for SMAP and MSL
- Ran Telemanom system over time period surrounding each ISA to see if system would have detected the anomaly



Results

Thresholding Approach	Precision	Recall	F_1 score
Non-Parametric w/ Pruning ($p = 0.1$)			
MSL	50.9%	63.6%	0.57
SMAP	62.6%	91.2%	0.74
Total	58.4%	80.4%	**0.68**

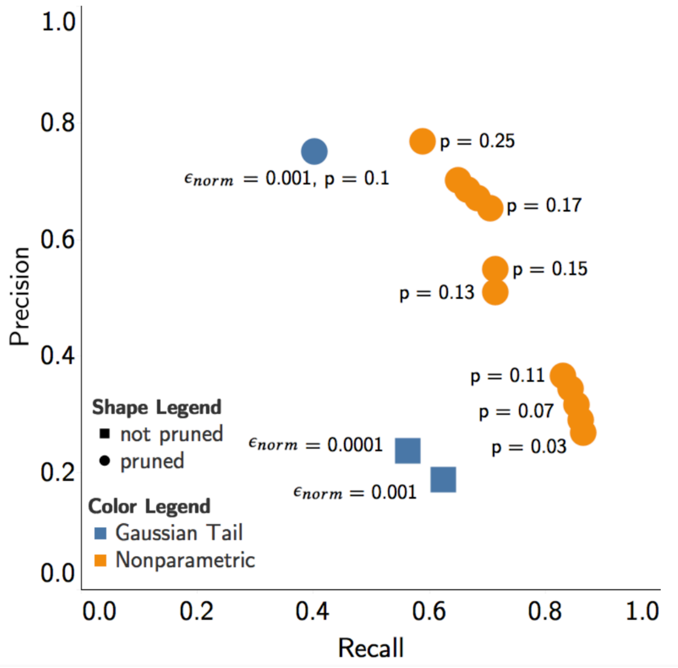
Over ½ of predicted anomalies were true positives

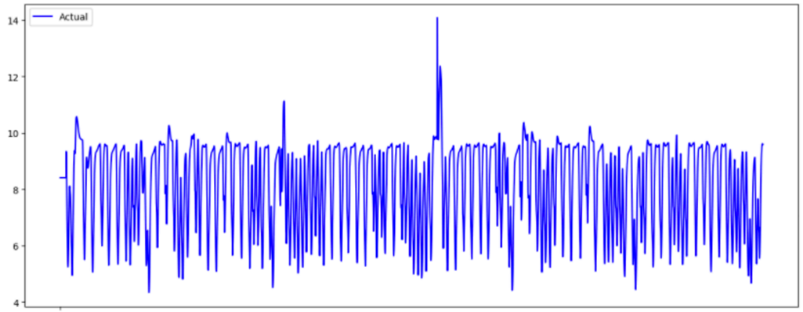
80% of all ISAs were identified (~115 in total)

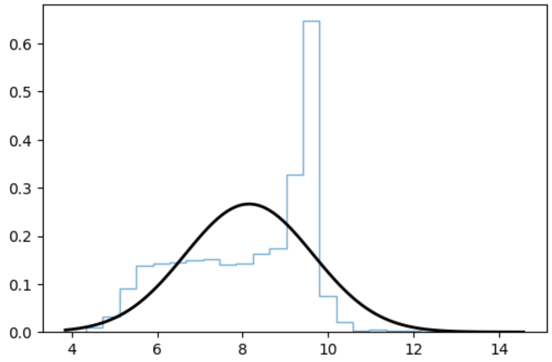
	Recall - *point*	Recall - *collective*
MSL	80.0%	42.1%
SMAP	97.7%	79.2%
Total	91.3%	62.8%

Collective anomalies are those that are not detectable by thresholds (0% recall)

Results





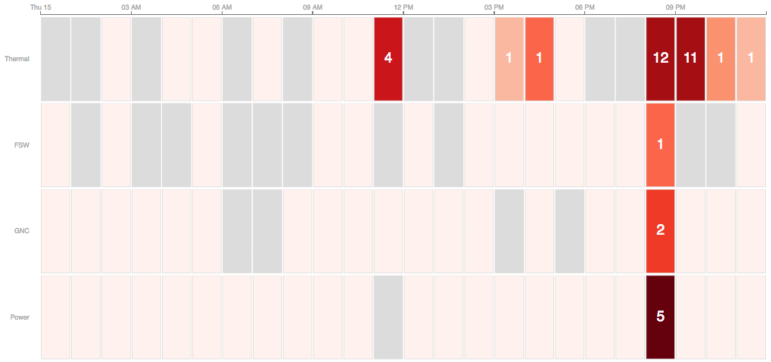


p-value in normal test: 1.17603615763e-237

Initial Pilot: SMAP

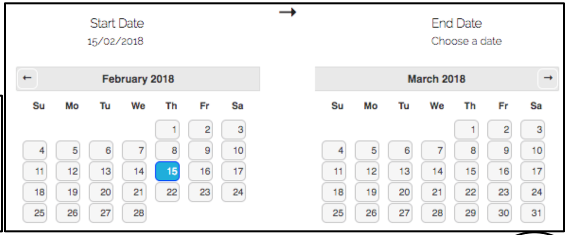
- Deployed end-to-end autonomous system
- Monitored ~750 core telemetry channels from Aug 2017 – May 2018
  - Detected multiple verified anomalous events
    - Partial eclipse (Feb 15, 2018)

- Radar (HPA) failure investigation
  - Ran system ~2 months prior to failure, detected many of same telemetry oddities that were identified during peer review process following failure



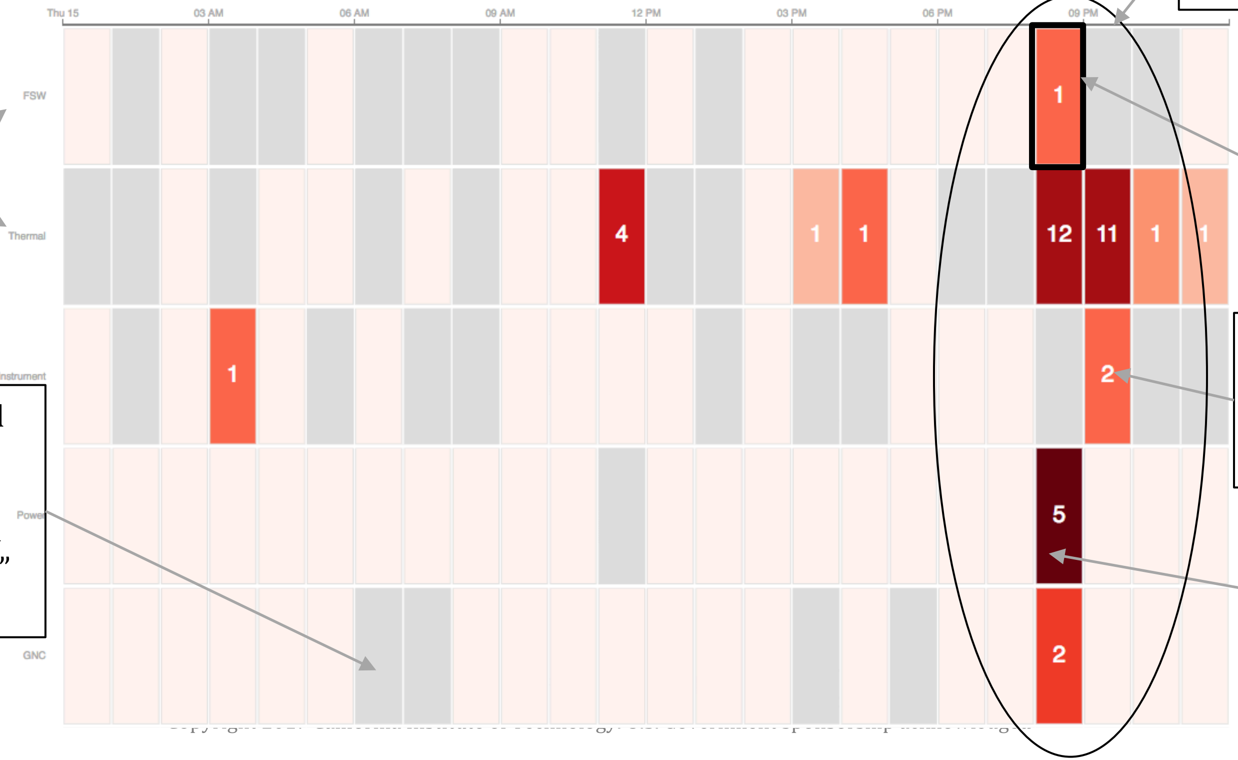
Interface: Top-Level Summary

Start by selecting a start and end date to look at



February 15th, 2018 Partial Solar Eclipse anomaly

Subsystem



Each box indicates one hour aggregate time window (adjustable)

Count of channel anomalies in subsystem during hour time window

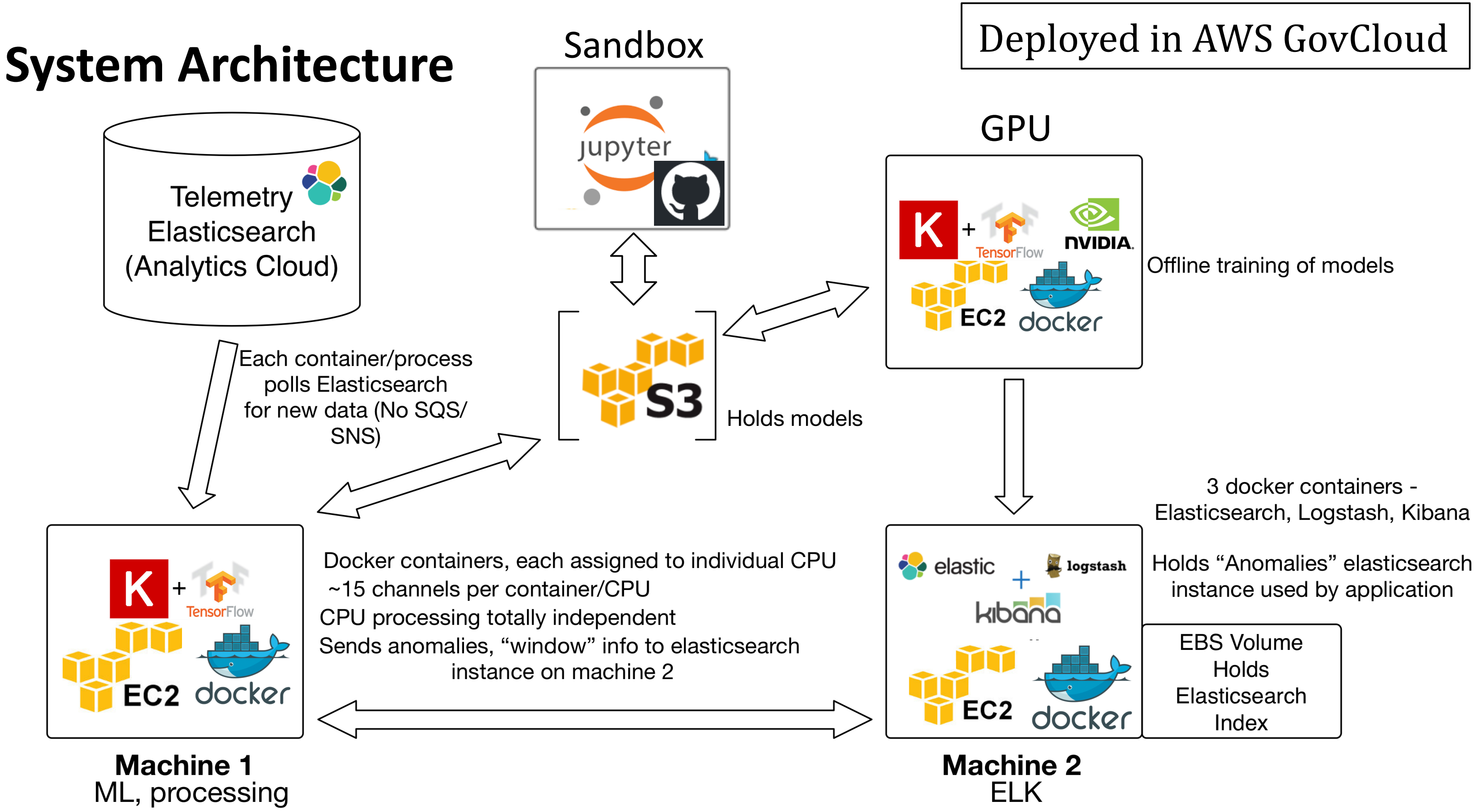
Gray boxes are potential anomalies that the system has learned are false positives with high likelihood ("suppressed" anomalies)

Darker color indicates

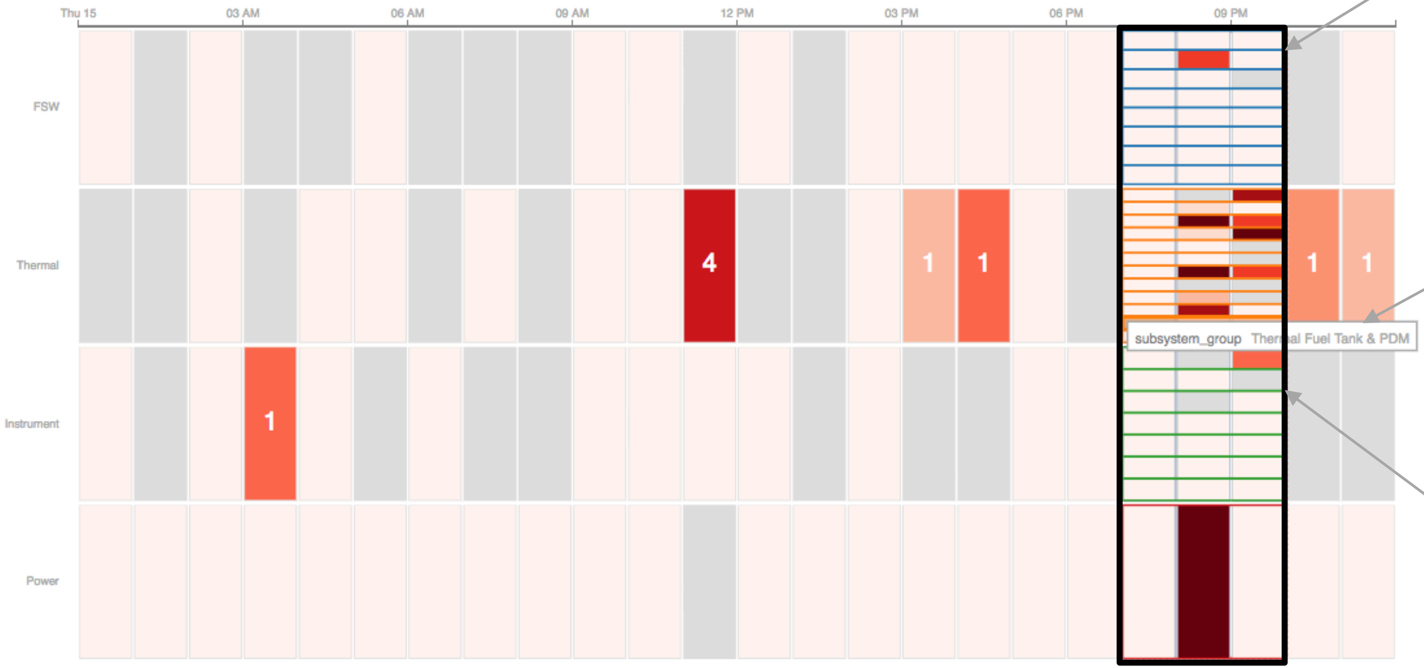
jpl.nasa.gov

Jet Propulsion Laboratory
California Institute of Technology

jpl.nasa.gov


System Architecture
Deployed in AWS GovCloud
Sandbox
jupyter
Telemetry
Elasticsearch
(Analytics Cloud)
GPU
K
+
TensorFlow
NVIDIA.
EC2
docker
Offline training of models
Each container/process
polls Elasticsearch
for new data (No SQS/
SNS)
S3
Holds models
3 docker containers -
Elasticsearch, Logstash, Kibana
K
+
TensorFlow
EC2
docker
Docker containers, each assigned to individual CPU
~15 channels per container/CPU
CPU processing totally independent
Sends anomalies, "window" info to elasticsearch
instance on machine 2
elastic
+
logstash
kibana
EC2
docker
Holds "Anomalies" elasticsearch
instance used by application
EBS Volume
Holds
Elasticsearch
Index
Machine 1
ML, processing
Machine 2
ELK

Interface: Drilldown

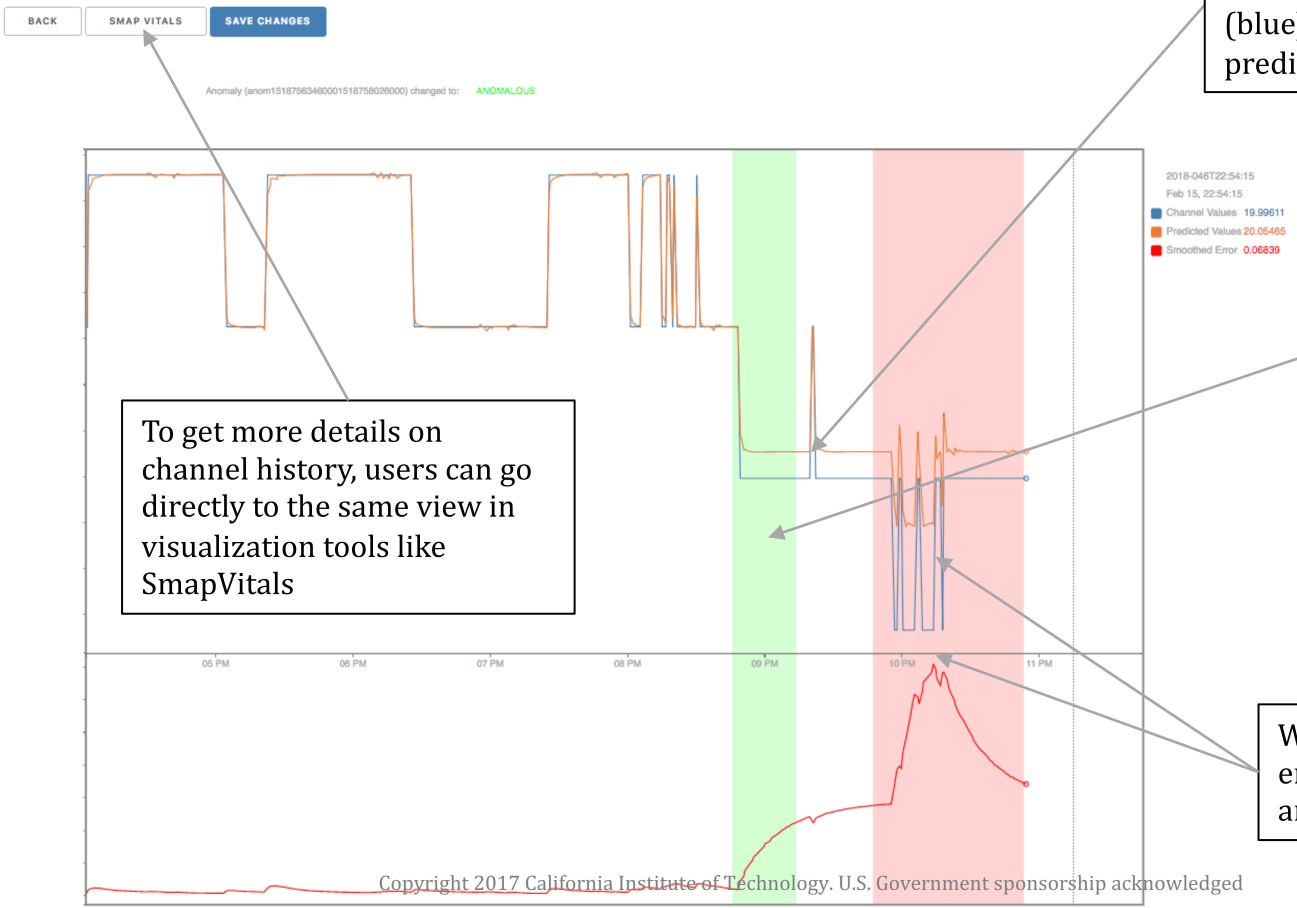


Clicking and dragging across an area allows for looking down a level to channel groups with subsystems

Each row represents a group of channels and hovering shows the group name

Clicking takes the user into a similar view but in the next level down for the selected window

Interface: Drilldown (cont.)



Users can drill down into the raw telemetry for each channel (blue) and compare to the model predictions (orange)

Users can click to tag anomalies as true or false positives, which are used by the system to refine results

True = green
False = gray
Unlabeled = red

To get more details on channel history, users can go directly to the same view in visualization tools like SmapVitals

Where prediction errors are large, anomalies are flagged

Current Work: MSL

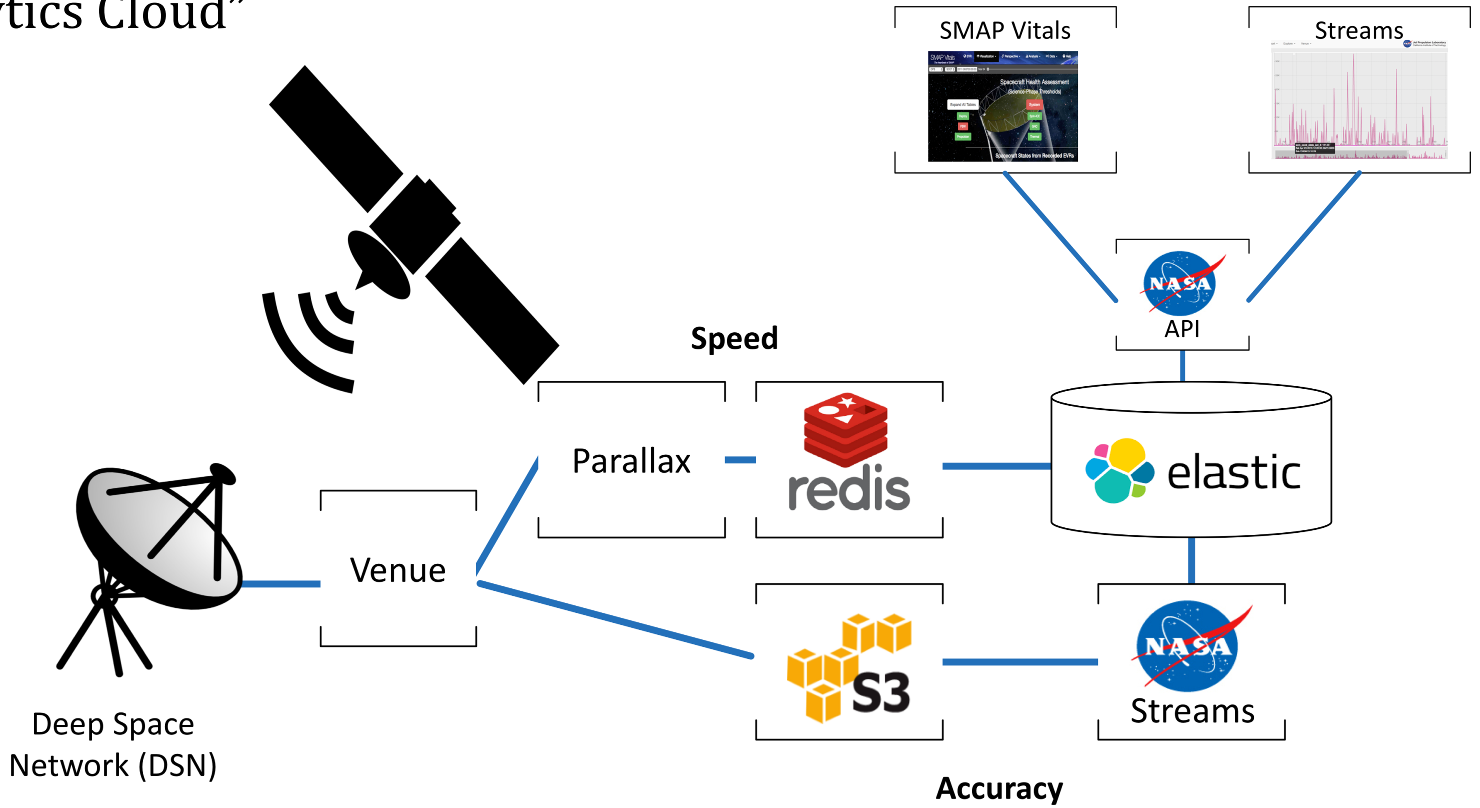
- Extending Telemanom to rovers/planetary missions
  - Prediction of telemetry is harder with more variety and irregularity of behaviors
  - Models need more training and detailed inputs surrounding commands and EVRs

- Targeting deployment of test system that will monitor Thermal, Power subsystems by end of FY2018

- Early progress
  - Detected Martian sandstorm early with small number of Thermal channels
  - Achieving very high prediction accuracy for thermal channels (~98%)

Foundation

The "Analytics Cloud"



Soil Moisture Active Passive (SMAP)

- Routine operations
- Major radar failure
- ~4,000 telemetry channels
  - Power, CPU, RAM, Thermal, Radiation, counters
  - 14 command modules
  - 4B values
- Challenges
  - Semi-supervised
  - Complexity, diversity
  - Scale